

Web Centralization



INTERPEER

@IETF115 // DINRG

Introduction

Interpeer Project does R&D in “internet technology”;

This talk is about the motivation;

Led to a grant from ISOC foundation to explore solutions.



<https://www.isocfoundation.org/>



Context

We can take an easy path or a hard path.

What happens when the easy path leads somewhere we don't want to end up?



Cable car & road at Tianmen mountain



Distribution

Distribution is about resilience; routing around failures is the goal.

Decentralization only provides this in the network at large, not for each centralized cluster.

Client/server paradigm opposes distribution.

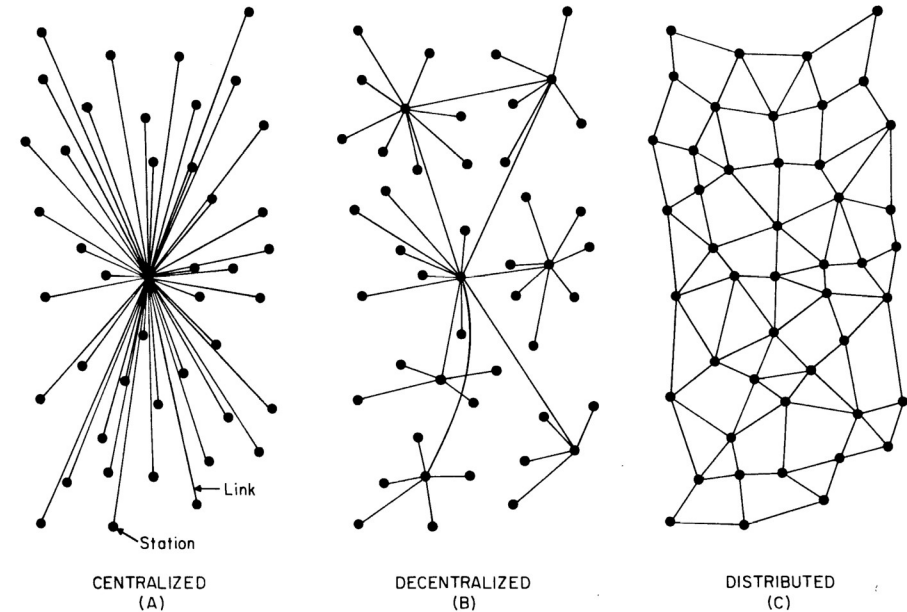


FIG. 1 – Centralized, Decentralized and Distributed Networks

Baran, P. (1964). On Distributed Communications, Memorandum RM-3420-PR.
https://www.rand.org/content/dam/rand/pubs/research_memoranda/2006/RM3420.pdf



What is Web? REST?

Fielding, Roy Thomas. Architectural Styles and the Design of Network-based Software Architectures. Doctoral dissertation, University of California, Irvine, 2000.

<https://www.ics.uci.edu/~fielding/pubs/dissertation/top.htm>

IS NOT HTTP (but inspired by work on HTTP/1.1)

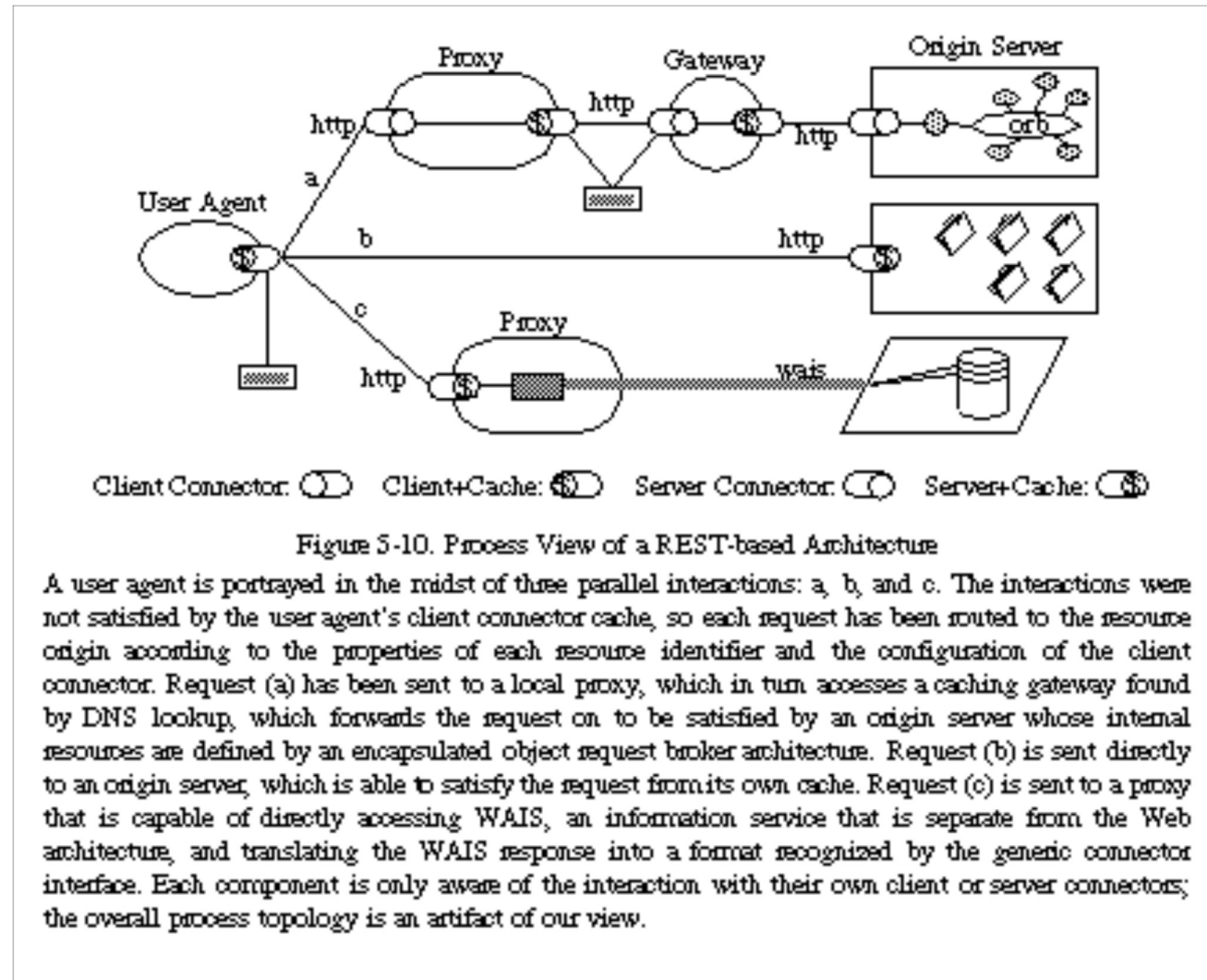
IS NOT software, framework, CRUD-style use of HTTP methods, etc.

RESTful has little to do with REST.

IS a decentralized, scalable architecture.



REST Overview



REST Overview

REST provides an interface; clients do not need to know how a service is implemented.

REST achieves this by defining that the data exchanged between client and service endpoint is a representation only.

This is an elegant method for enabling a generic client.



Web Centralization

HTTP is inherently client/server based; this pre-assigns roles to nodes and prevents nodes from “routing around failures”.

AAA on the web is centralized (in a service); the endpoint decides, not the end user. This may also be a single point of failure for a service.

RESTful may mistakenly map CRUD-style operations onto HTTP methods, but...



Web Centralization / CRUD

PUT works as a generic CREATE; GET as a generic READ; DELETE as a generic DELETE;

POST is terrible as a generic UPDATE:

- Representational means the client cannot a priori know how to format a request.
- Requires the server to send appropriate code to the client (optional in REST).
- Creates data + code silos in services.



Web Centralization / Economic Incentives

Service providers control access via AAA, control data via de-facto mandatory code for modification.

Service providers can monetize this better the more data they gather, which means the incentives are stacked towards platformization (centralization).

Surveillance capitalism is a natural fit for this.



Web Centralization

Centralization isn't required by the web, but the incentives are stacked towards it.

How can we make the easy path also the path that pushes towards distribution?



Cable car & road at Tianmen mountain



Distribution / AAA

Move AAA from service endpoints to end users.

- Distributed authentication: “solved” (the hard part, via public key cryptography).
- Distributed authorization: capabilities, power of attorney (?)

There is plenty of room for improvement, but it’s far from impossible.



Distribution / E2E data + operations

Make data representation and operations an end-to-end problem (client-to-client)...

E2EE takes out the service as interfering with the process; have data routers instead of services.

CRUD style operations are no longer necessary at the transport; in fact, HTTP is no longer necessary.

Natural fit for Information-Centric Networking (ICNRG).





THANK YOU

 **INTERPEER**